UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/756,863	01/10/2001	Toyohiko Yoshida	49657-921	6032

7590 09/09/2004

McDERMOTT, WILL & EMERY
600 13th Street, N.W.
Washington, DC 20005-3096

EXAMINER	
O#BRIEN, BARRY J	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 09/09/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/756,863	YOSHIDA, TOYOHIKO	
	Examiner	Art Unit	
	Barry J. O'Brien	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *12 August 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-12,14,16,18 and 19* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-12,14,16,18 and 19* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-12, 14, 16, 18 and 19 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed on record in the file: Amendment After Final as received on 8/12/2004.

Specification

3. The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

4. The applicant is requested to review the specification and update the status of all co-pending applications made mention of, replacing attorney docket numbers with current U.S. application or patent numbers when appropriate.

Claim Objections

5. Claims 1 and 12 are objected to because of the following informalities:

   a. Claim 1 recites the limitation, "an instruction output by said translator and the corresponding instruction held in said instruction cache". Please change the limitation to read, "an instruction output by said translator or the corresponding instruction held in said instruction cache" so as to more clearly point out the items being selected. See claim 12 for a similar required correction.

Appropriate correction is required.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 1, 5, 12, 14, 16 and 18-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Emma, U.S. Patent No. 5,619,665, in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (hereinafter IBM).

8. Regarding claim 1, Emma has taught in a processor (508 of Fig.5) operating with instructions in a first instruction architecture as a native instruction, an instruction translator (506 of Fig.5) used with an instruction memory (500 of Fig.5) to store an instruction in a second instruction architecture different from said first instruction architecture, for translating an instruction in said second instruction architecture into an instruction in said first instruction architecture for application to said processor (see Col.3 lines 2-17), said instruction translator comprising;

a. A translator (506 of Fig.5) for reading out an instruction from said instruction memory (500 of Fig.5) in response to a received first address in said instruction memory of an instruction to be executed by said processor and translating the read out instruction in said second instruction architecture into an instruction in said first instruction architecture (see Col.12 lines 1-23),

b. An instruction cache (504 of Fig.5) for temporarily holding the instruction in said first instruction architecture after the translation by said translator in association with the first address in said instruction memory (see Col.12 lines 17-23).

9. Emma has not explicitly taught a selector for searching said instruction cache in response to a received second address of an instruction to be executed by said processor, and for selectively outputting, based on a determination result of whether or not an instruction corresponding to the instruction of the second address is held in said instruction cache, an instruction output by said translator and the corresponding instruction held in said instruction cache.

10. However, IBM has taught the selection of either an instruction stored in an instruction cache or an instruction from the main memory store based on if there is a cache miss (see p.1 lines 10-15 and p.2 lines 2-7) so as to reduce the delay associated with waiting for an instruction while a cache line is being retrieved on a cache miss (see p.1 lines 1-9). One of ordinary skill in the art would have recognized that the instructions from the main memory store that are to be stored in the cache containing translated instructions (504 of Fig.5) have to come from the translator prior to being stored. Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Emma to include a cache bypass around the EI-Cache of Emma so that instructions can be selected from either the output of the translator or the EI-Cache based on a cache hit/miss in order to reduce the delay associated with a fetch to memory on an instruction cache miss.

11. Regarding claim 5, Emma in view of IBM has taught the instruction translator according to claim 1, wherein said translator includes a plurality of translators which translate a plurality of

instruction in said second instruction architecture read out from said instruction memory into one instruction in said first instruction architecture (see Emma, Col.3 lines 2-17).

12. While a plurality of translators is not explicitly taught, the translation of a plurality of instructions in the second instruction architecture into one instruction in the first instruction architecture is taught. The inclusion of a plurality of translators to perform the same function as a single translator provides no new or unexpected result over the prior art. Therefore, one of ordinary skill in the art would have found it obvious to duplicate the translator, creating a plurality of translators for translating instructions in a second instruction architecture into a single instruction in a first instruction architecture (see In re Harza, 274 F.2d 669, 671, 124 USPQ 378, 380 (CCPA 1960)).

13. Regarding claim 12, Emma has taught an instruction memory attached with a translator, used with a processor operating with an instruction in a first instruction architecture as a native instruction (see Col.3 lines 2-17), comprising:

a. An instruction storage unit (500 of Fig.5) to store an instruction in a second instruction architecture (see Col.12 lines 1-23),

b. An instruction translator (506 of Fig.5) to translate an instruction in said second instruction architecture output from said instruction storage unit into an instruction in said first instruction architecture for application to said processor (see Col.12 lines 1-23), said instruction translator including:

I. A translator (506 of Fig.5) to read out the instruction in said second instruction architecture from said instruction storage unit (500 of Fig.5) in response to a received first address of an instruction to be executed by said

processor and translate the read out instruction in said second instruction architecture in the instruction in said first instruction architecture (see Col.12 lines 1-23),

II. An instruction cache (504 of Fig.5) to temporarily hold the instruction in said first instruction architecture after the translation by said translator in association with the first address (see Col.12 lines 17-23).

14. Emma has not explicitly taught a selector to search said instruction cache in response to a received second address of an instruction to be executed by said processor and selectively output to said processor, based on a determination result of whether or no an instruction corresponding to the instruction of the address is held in said instruction cache, and instruction output by said translator and the corresponding instruction in said first instruction architecture held in said instruction cache.

15. However, IBM has taught the selection of either an instruction stored in an instruction cache or an instruction from the main memory store based on if there is a cache miss (see p.1 lines 10-15 and p.2 lines 2-7) so as to reduce the delay associated with waiting for an instruction while a cache line is being retrieved on a cache miss (see p.1 lines 1-9). One of ordinary skill in the art would have recognized that the instructions from the main memory store that are to be stored in the cache containing translated instructions (504 of Fig.5) have to come from the translator prior to being stored. Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Emma to include a cache bypass around the EI-Cache of Emma so that instructions can be selected from either the output of the translator or the EI-Cache

based on a cache hit/miss in order to reduce the delay associated with a fetch to memory on an instruction cache miss.

16. Regarding claim 14, Emma in view of IBM has taught the instruction memory attached with a translator according to claim 12, further comprising an address translator to translate an address at the time of reading from said instruction storage unit (see Emma, Col.11 lines 4-13). Here, Emma has taught the address of source instructions A, B, and C being translated to a native instruction X_1 having the address of source instruction A, the translating the addresses of B and C to be the address of A (X_1).

17. Regarding claim 16, Emma has taught a data processing apparatus, comprising:

a. A processor (508 of Fig.5) operating with an instruction in a first instruction architecture as a native instruction (see Col.3 lines 2-17),

b. A bus (510 of Fig.5) to which said processor is connected,

c. A first instruction memory (500 of Fig.5) with a translator (506 of Fig.5) interconnected with said processor through said bus (see Fig.5),

d. A second instruction memory (504 of Fig.5) interconnected to said processor through said bus (see Fig.5),

e. Said first instruction memory with a translator including:

I. A first instruction storage unit (500 of Fig.5) to store an instruction in a second instruction architecture transferred from said processor through said bus (see Col.12 lines 1-9),

II. An instruction translator (506 of Fig.5) to translate the instruction in said second instruction architecture output from said first instruction storage

unit into an instruction in said first instruction architecture for application to said processor through said bus (see Col.12 lines 1-23),

f. Said second instruction memory including:

I. A second instruction storage unit (504 of Fig.5) to store an instruction in said first instruction architecture transferred from said processor through said bus (see Col.12 lines 17-23).

18. Emma has not explicitly taught an instruction reading circuit responsive to an address signal applied form said processor through said bus for applying an instruction in said first instruction architecture output form said second instruction storage unit to said processor through said bus.

19. However, IBM has taught the selection of either an instruction stored in an instruction cache or an instruction from the main memory store based on if there is a cache miss (see p.1 lines 10-15 and p.2 lines 2-7) so as to reduce the delay associated with waiting for an instruction while a cache line is being retrieved on a cache miss (see p.1 lines 1-9). One of ordinary skill in the art would have recognized that the instructions from the main memory store that are to be stored in the cache containing translated instructions (504 of Fig.5) have to come from the translator prior to being stored. Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Emma to include a cache bypass around the EI-Cache of Emma so that instructions can be selected from either the output of the translator or the EI-Cache based on a cache hit/miss in order to reduce the delay associated with a fetch to memory on an instruction cache miss.

20. Regarding claim 18, Emma in view of IBM has taught the data processing apparatus according to claim 16, wherein the number of clock cycles spent by said processor to access said first instruction memory through said bus is larger than the number of clock cycles spent by said processor to access said second instruction memory through said bus (see IBM, p.1 lines 1-9 and p.2 lines 2-7).

21. Regarding claim 19, Emma in view of IBM has taught the data processing apparatus according to claim 16, further comprising a second instruction memory (see Emma, 504 of Fig.5) with a translator interconnected to said processor through said bus, said second instruction memory with a translator including:

a. An instruction storage unit (see Emma, 504 of Fig.5) to store an instruction in a second instruction architecture different from said second instruction architecture, said instruction in said third instruction architecture being transferred from said processor through said bus (see Emma, Col.12 lines 1-23),

b. An instruction translation circuit (see Emma, 506 of Fig.5) responsive to an address signal applied from said processor through said bus, for translating an instruction in said third instruction architecture output from said instruction storage unit into an instruction in said first instruction architecture for application to said processor (see Emma, Col.12 lines 1-35).

22. Emma in view of IBM has not explicitly taught a third instruction memory with a translator for translating a third instruction architecture into a first instruction architecture is not explicitly taught.

23. However, adding a third instruction memory and corresponding third instruction architecture provides no new or unexpected result over the prior art except for allowing the additional translation of a third instruction architecture. But this could be achieved simply by replacing the second instruction architecture with the third architecture, as there are no claim limitations that require the two instruction architectures to be translated simultaneously. Therefore, one of ordinary skill in the art would have found it obvious to duplicate the hardware used to operate on a second instruction architecture as taught by Emma in view of IBM in order to operate on a third instruction architecture (see In re Harza, 274 F.2d 669, 671, 124 USPQ 378, 380 (CCPA 1960)).

24. Claims 2-3 are rejected under 35 U.S.C. 103(a) as being unpatentable over Emma, U.S. Patent No. 5,619,665, in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (hereinafter IBM), as applied to claim 1 above, and further in view of Dickol et al., U.S. Patent No. 5,875,336.

25. Regarding claim 2, Emma in view of IBM has taught the instruction translator according to claim 1, but has not explicitly taught wherein said second instruction architecture is a variable length instruction architecture, and said translator includes a variable length translator for translating an instruction in said second instruction architecture read out from said instruction memory into one or more instruction in said first instruction architecture, the number of which depends on an instruction length of the read out instruction in said second instruction architecture.

26. However, Dickol has taught a variable length instruction architecture being translated in real time into one or more instructions of a second instruction architecture based on the varying

lengths of instructions in the variable length instruction architecture (see Dickol, Col.3 lines 29-38 and Col.4 lines 13-14, 21-27) in order to improve performance when executing non-native instructions on more common native-instruction-based hardware (see Dickol, Col.1 lines 47-61 and Col.2 lines 15-18). One of ordinary skill in the art would have recognized that a priority of microprocessor designers is to improve performance while minimizing cost and hardware complexity. Therefore, one of ordinary skill in the art at the time of the invention would have found it obvious to modify Emma in view of IBM to further decode instructions from a non-native variable length instruction set into one or more native instructions in order to improve processor performance when executing non-native instructions while minimizing the additional hardware required to do so.

27. Regarding claim 3, Emma in view of IBM in further view of Dickol has taught the instruction translator according to claim 2, wherein said variable length translator translates the instruction in said second instruction architecture read out from said instruction memory into an instruction in said first instruction architecture having a total length depending on and larger than the instruction length of said read out instruction in said second instruction architecture (see Dickol, Col.3 lines 29-38 and Col.4 lines 13-14, 21-27).

28. Claim 4 is rejected under 35 U.S.C. 103(a) as being unpatentable over Emma, U.S. Patent No. 5,619,665 in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (hereinafter IBM) in view of Dickol et al., U.S. Patent No. 5,875,336 as applied to claims 1-3 above, and further in view of Goettelmann et al., U.S. Patent No. 5,313,614.

29. Regarding claim 4, Emma in view of IBM in view of Dickol have taught the instruction translator according to claim 3, but have not explicitly taught wherein each instruction in said

first instruction architecture includes one or a plurality of sub instructions, and the number of the sub instructions included in the instruction in the first instruction architecture translated by said variable length translator depends on the instruction length of said read out instruction in said second instruction architecture.

30. However, Goettelmann has taught the translation from a source instruction architecture (see Goettelmann, "source machine code" of Fig.11) into a native instruction architecture (see "translated code" of Fig.11), wherein the native architecture contains a plurality of sub-instructions (see Goettelmann, "expanded intermediate language code" of Fig.11), sub-instructions which can then be removed if necessary in order to reduce the translated code size (see Goettelmann, Col.4 lines 6-18). One of ordinary skill in the art would have recognized that it is desirable to reduce the size of instruction code so that less hardware (memory space) is needed, thus lowering costs. Therefore, one of ordinary skill in the art would have found it obvious to modify the instruction translator of Emma in view of IBM in view of Dickol to further translate instructions from a source architecture into a native architecture that includes a plurality of sub-instructions in order to reduce the amount of instruction storage required to store the translated code. Furthermore, because the number of native instructions are dependent on the length of the varying-length source instruction as shown above, and because the number of sub instructions depend on the native instructions, then the number of sub-instructions depend on the length of the source instructions.

31. Claims 6-7 are rejected under 35 U.S.C. 103(a) as being unpatentable over Emma, U.S. Patent No. 5,619,665 in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass*

During Cache Reload (hereinafter IBM) as applied to claim 1 above, and further in view of Goettelmann et al., U.S. Patent No. 5,313,614.

32. Regarding claim 6, Emma in view of IBM has taught the instruction translator according to claim 1, but have not explicitly taught wherein:

a. Each instruction in said first instruction architecture can include one or a plurality of sub instructions,

b. Said translator translates a plurality of instructions in said second instruction architecture read out from said instruction memory into an instruction in said first instruction architecture including sub instructions, the number of which depends on the number of said plurality of instructions.

33. However, Goettelmann has taught the translation from a source instruction architecture (see Goettelmann, "source machine code" of Fig.11) into a native instruction architecture (see Goettelmann, "translated code" of Fig.11), wherein the native architecture contains a plurality of sub-instructions (see Goettelmann, "expanded intermediate language code" of Fig.11), sub-instructions which can then be removed if necessary in order to reduce the translated code size (see Goettelmann, Col.4 lines 6-18). One of ordinary skill in the art would have recognized that it is desirable to reduce the size of instruction code so that less hardware (memory space) is needed, thus lowering costs. Therefore, one of ordinary skill in the art would have found it obvious to modify the instruction translator of Emma in view of IBM to further translate instructions from a source architecture into a native architecture that includes a plurality of sub-instructions in order to reduce the amount of instruction storage required to store the translated code. Furthermore, because the number of native instructions are dependent on the length of the

varying-length source instruction as shown above, and because the number of sub instructions depend on the native instructions, then the number of sub-instructions depend on the length of the source instructions.

34. Regarding claim 7, Emma in view of IBM in further view of Goettelmann have taught the instruction translator according to claim 6, wherein the number of sub instructions included in the instruction in said first instruction architecture after said translation is equal to the number of said plurality of instructions (see "BEQ Label instruction" of "source machine code", its corresponding "BFALSE FlagZ, Label" instruction of "translated code", with associated sub-instruction "BFALS.d FlagZ, Label" of "expanded intermediate language code", all of Fig.11 of Goettelmann).

35. Claims 8 and 10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Emma, U.S. Patent No. 5,619,665 in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (hereinafter IBM), further in view of Gregor, U.S. Patent No. 5,023, 776.

36. Regarding claim 8, Emma in view of IBM has taught the instruction translator according to claim 1 as shown above, wherein:

a. Said translator translates said read out instruction in said second instruction architecture into one or a plurality of instructions as the instruction in said first instruction architecture (see Emma, Col.12 lines 1-23)

37. Emma in view of IBM has not explicitly taught where said instruction translator further comprises a controller for controlling said instruction cache so that said instruction cache holds

each of said one or said plurality of instruction held in said instruction cache as an entry which can be invalidated in one of first and second conditions.

38. However, Gregor has taught the holding of a cache line within a cache so that it cannot be replaced until an EOP signal is detected in order to reduce cache busy time and improve processor performance (see Gregor, Col.22 lines 36-67). One of ordinary skill in the art would have recognized that it is desirable, as well as a main goal of microprocessor design, to improve the performance of a microprocessor. Therefore, one of ordinary skill in the art would have found it obvious to modify the cache of Emma in view of IBM to hold a plurality of instructions in a cache until an EOP signal is detected. Furthermore, because the claim language is in the alternative format, only one of the two invalidation condition requirements is required to be met, and thus the EOP signal that is detected (see Gregor, Col.22 lines 56-57) can be considered such a signal.

39. Regarding claim 10, Emma in view IBM in further view of Gregor has taught the instruction translator according to claim 8, wherein said controller outputs a signal asserted when a new instruction cannot be held in said instruction cache without invalidating an entry which can be invalidated in the second condition (see Gregor, Col.22 lines 36-67).

40. Claims 9 and 11 are rejected under 35 U.S.C. 103(a) as being unpatentable over Emma, U.S. Patent No. 5,619,665 in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (hereinafter IBM) in view of Gregor, U.S. Patent No. 5,023, 776, in further view of Schacham et al., U.K. Patent Application GB220481A.

41. Regarding claim 9, Emma in view of IBM in view of Gregor have taught the instruction translator according to claim 8, wherein:

a. Said first condition is a holding control condition by hardware control based on a prescribed algorithm by said instruction cache (see Gregor, Col.22 lines 36-67).

42. Emma in view of IBM in further view of Gregor has not explicitly taught wherein the said second condition is a condition in which an explicit invalidation instruction is applied from the outside of said instruction cache.

43. However, in the parent claim of claim 9 only one of the two invalidation conditions must be met. Therefore, because Emma in view of IBM in further view of Gregor has satisfied one of the conditions, namely the holding control condition as shown above, the claim language is satisfied.

44. Furthermore, even though the alternative form claim language is met, Schacham has taught that a cache invalidation instruction can be executed to invalidate the entire instruction cache or a portion of it (see Schacham, p.4 lines 28-30) in order to maintain proper cache coherence and provide better processor performance (see Schacham, p.2 lines 8-23). One of ordinary skill in the art would have recognized that it is desirable, as well as a main goal of microprocessor design, to improve the performance of a microprocessor. Therefore, one of ordinary skill in the art would have found it obvious to modify the instruction cache of Emma in view of IBM in view of Gregor to allow a cache invalidation instruction to be executed so that cache coherency is maintained and processor performance is improved.

45. Regarding claim 11, Emma in view of IBM in view of Gregor has taught the instruction translator according to claim 8, wherein said translator translates the read out instruction in said second instruction architecture into the plurality of instructions as the instruction in said first instruction architecture (see Emma, Col.12 lines 1-23), but has not explicitly taught where said

controller provides said first condition with one of said plurality of instructions and said second condition with each of said plurality of instructions but said one instruction.

46. However, Schacham has taught that a cache invalidation instruction can be executed to invalidate the entire instruction cache or a portion of it (see Schacham, p.4 lines 28-30) in order to maintain proper cache coherence and provide better processor performance (see Schacham, p.2 lines 8-23). One of ordinary skill in the art would have recognized that it is desirable, as well as a main goal of microprocessor design, to improve the performance of a microprocessor. Therefore, one of ordinary skill in the art would have found it obvious to modify the instruction cache of Emma in view of IBM in view of Gregor to allow a cache invalidation instruction to be executed so that cache coherency is maintained and processor performance is improved.

47. Furthermore, Emma in view of IBM in further view of Gregor in view of Schacham has taught the invalidation of instructions in a cache based upon one of two conditions, namely a holding control condition as taught by Gregor, and a cache invalidation instruction as taught by Schacham. One of ordinary skill in the art would have recognized that because some instructions will have the first condition associated with them, and others will have the second condition associated with them. Therefore, one of ordinary skill in the art would have found it obvious that the processor of Emma in view of IBM in further view of Gregor in view of Schacham will provide one instruction with a first condition, and other instructions with a second condition.

Response to Arguments

48. Applicant's arguments with respect to claims 1-12, 14, 16, 18 and 19 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

49. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is reminded that in amending in response to a rejection of claims, the patentable novelty must be clearly shown in view of the state of the art disclosed by the references cited and the objections made. Applicant must also show how the amendments avoid such references and objections. See 37 CFR § 1.111(c).

50. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Barry J. O'Brien whose telephone number is (703) 305-5864. After October 12th, 2004, the examiner can be reached at (571) 272-4171. The examiner can normally be reached on Mon.-Fri. 6:30am-4:00pm, with the exception of first Friday of every bi-week.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached at (703) 305-9712, or at (571) 272-4162 on or after October 12th, 2004. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

51. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Barry J. O'Brien
Examiner
Art Unit 2183

BJO
9/7/2004

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100